

## CLAIMS

1           1. (Previously Presented) A processing system for performing addition and  
2 subtraction within limits upon a shared value comprising:  
3           means for performing a first uninterruptible operation upon the shared value  
4 stored in an affected reservation location, the first uninterruptible operation using an  
5 operand;  
6           means for comparing a resulting value of the first uninterruptible operation stored  
7 in the affected reservation location to an upper value and a lower value to determine if the  
8 resulting value is within a range defined by the upper value and the lower value that can  
9 be changed;  
10          means for performing a second uninterruptible operation to restore the affected  
11 reservation location if the resulting value of the first uninterruptible operation is not  
12 within the range defined by the upper value and the lower value;  
13          means for reporting a failure if the resulting value of the first uninterruptible  
14 operation is not within the range defined by the upper value and the lower value;  
15          means for performing a third uninterruptible operation to update an actual value  
16 location if the resulting value of the first uninterruptible operation is within the range  
17 defined by the upper value and the lower value;  
18          means for performing a fourth uninterruptible operation to update an unaffected  
19 reservation location if the resulting value of the first uninterruptible operation is within  
20 the range defined by the upper value and the lower value; and  
21          means for reporting a success if the resulting value of the first uninterruptible  
22 operation is within the range defined by the upper value and the lower value.

1           2. (Previously presented) The processing system of claim 1 wherein the first,  
2 second, third, and fourth uninterruptible operations are LOCK XADD operations.

1           3. (Currently Amended) A processing system for performing addition and  
2 subtraction within limits upon a shared value comprising:

3                   means for receiving an operand;

4                   means for performing a first uninterruptible operation upon the shared  
5 value stored in an affected reservation location, the first uninterruptible operation  
6 using the operand;

7                   means for comparing a resulting value of the first uninterruptible operation  
8 stored in the affected reservation location to an upper value and a lower value to  
9 determine if the resulting value is within a range defined by the upper value and  
10 the lower value that can be changed;

11                   means for performing a second uninterruptible operation to restore the  
12 affected reservation location if the resulting value of the first uninterruptible  
13 operation is not within the range defined by the upper value and the lower value;

14                   means for ~~and~~ reporting a failure if the resulting value of the first  
15 uninterruptible operation is not within the range defined by the upper value and  
16 the lower value;

17                   means for performing a third uninterruptible operation to update an actual  
18 value location if the resulting value of the first uninterruptible operation is within  
19 the range defined by the upper value and the lower value;

20 means for performing a fourth uninterruptible operation to update an  
21 unaffected reservation location if the resulting value of the first uninterruptible  
22 operation is within the range defined by the upper value and the lower value; and  
23 means for reporting a success if the resulting value of the first  
24 uninterruptible operation is within the range defined by the upper value and the  
25 lower value.

1 4. (Previously presented) The processing system of claim 3 wherein the first,  
2 second, third, and fourth uninterruptible operations are LOCK XADD operations.

1 5. (Previously Presented) A method for performing addition and subtraction  
2 within limits upon a shared value comprising the steps of:

3 first, performing a first uninterruptible operation upon the shared value  
4 stored in an affected reservation location, the first uninterruptible operation using  
5 an operand;

6 second, comparing a resulting value of the first uninterruptible operation  
7 stored in the affected reservation location to an upper value and a lower value to  
8 determine if the resulting value is within a range defined by the upper value and  
9 the lower value that can be changed;

10 third, performing a second uninterruptible operation to restore the affected  
11 reservation location;

12 fourth, reporting a failure if the resulting value is not within the range  
13 defined by the upper value and the lower value;

14 fifth, performing a third uninterruptible operation to update an actual value  
15 location if the resulting value is within the range defined by the upper value and  
16 the lower value;

17 sixth, performing a fourth uninterruptible operation to update an  
18 unaffected reservation location if the resulting value is within the range defined  
19 by the upper value and the lower value; and

20 seventh, reporting a success if the resulting value is within the range  
21 defined by the upper value and the lower value.

1 6. (Previously Presented) The method of claim 5 wherein the first, second, third,  
2 and fourth uninterruptible operations are LOCK XADD operations.

1 7. (Previously Presented) A computer readable medium containing computer  
2 readable code comprising:

3 a code segment for performing a first uninterruptible operation upon the  
4 shared value stored in an affected reservation location, the first uninterruptible  
5 operation using an operand;

6 a code segment for comparing a resulting value of the first uninterruptible  
7 operation stored in the affected reservation location to an upper value and a lower  
8 value to determine if the resulting value is within a range defined by the upper  
9 value and the lower value that can be changed;

10 a code segment for performing a second uninterruptible operation to  
11 restore the affected reservation location;

12                   a code segment for reporting a failure if the resulting value is not within  
13                   the range defined by the upper value and the lower value;  
14                   a code segment for performing a third uninterruptible operation to update  
15                   an actual value location if the resulting value is within the range defined by the  
16                   upper value and the lower value;  
17                   a code segment for performing a fourth uninterruptible operation to update  
18                   an unaffected reservation location if the resulting value is within the range defined  
19                   by the upper value and the lower value; and  
20                   a code segment for reporting a success if the resulting value is within the  
21                   range defined by the upper value and the lower value.

1           8.       (Previously Presented)       A processing system for performing addition  
2   and subtraction within limits upon a shared value comprising:  
3                   a processor, the processor  
4                   performing a first uninterruptible operation upon the shared value  
5                   stored in an affected reservation location, the first uninterruptible  
6                   operation using an operand;  
7                   comparing a resulting value of the first uninterruptible operation  
8                   stored in the affected reservation location to an upper value and a lower  
9                   value to determine if the resulting value is within a range defined by the  
10                  upper value and the lower value that can be changed;  
11                  performing a second uninterruptible operation to restore the  
12                  affected reservation location if the resulting value of the first

13                   uninterruptible operation is not within the range defined by the upper  
14                   value and the lower value;  
15                   reporting a failure if the resulting value of the first uninterruptible  
16                   operation is not within the range defined by the upper value and the lower  
17                   value;  
18                   performing a third uninterruptible operation to update an actual  
19                   value location if the resulting value of the first uninterruptible operation is  
20                   within the range defined by the upper value and the lower value;  
21                   performing a fourth uninterruptible operation to update an  
22                   unaffected reservation location if the resulting value of the first  
23                   uninterruptible operation is within the range defined by the upper value  
24                   and the lower value; and  
25                   reporting a success if the resulting value of the first uninterruptible  
26                   operation is within the range defined by the upper value and the lower  
27                   value.

1           9. (New) A method of performing subtraction or addition within limits, the  
2   method comprising:

3                   receiving an operand from a processing thread;  
4                   performing a first uninterruptible operation upon an affected reservation  
5                   location, the affected reservation location including a first instance of a shared  
6                   first value, the first uninterruptible operation being configured to generate a  
7                   second value by subtracting the operand from or adding the operand to the first  
8                   instance of the shared first value;

9 comparing the generated second value in the affected reservation location  
10 to one or more limit values stored in one or more limit locations;

11 performing a second uninterruptible operation to restore the shared first  
12 value in the affected reservation location if the second value is not within any of  
13 the one or more limit values;

14 reporting a failure if the second value is not within any of the one or more  
15 limit values;

16 performing a third uninterruptible operation to update a second instance of  
17 the shared first value stored in an actual value location if the second value is  
18 within the one or more limit values, the actual value location being a memory  
19 location shared by a plurality of processing threads; and

20 performing a fourth uninterruptible operation to update a third instance of  
21 the shared first value stored in an unaffected reservation location if the second  
22 value is within the one or more limit values.

1 10. (New) The method of claim 9, further including reporting a success if the  
2 second value is within the one or more limit values.

1 11. (New) The method of claim 9, wherein the one or more limit values consist  
2 of an upper limit value and a lower limit value.

1 12. (New) The method of claim 9, wherein the first uninterruptible operation is  
2 configured to generate the second value by adding the operand to the first instance of the  
3 shared first value.

1           13. (New) The method of claim 9, wherein the first uninterruptible operation is  
2 configured to generate the second value by subtracting the operand from the first instance  
3 of the shared first value.

1           14. (New) The method of claim 9, wherein the operand has an absolute value  
2 greater than one.

1           15. (New) The method of claim 9, wherein performing the second  
2 uninterruptible operation includes using a negative of the operand.

1           16. (New) The method of claim 9, further including choosing the first affected  
2 reservation location on which to perform the first uninterruptible operation, responsive to  
3 whether the method is being used to perform a subtraction or addition.

          17. (New) The system of claim 1, further including means configured for  
5 choosing the first affected reservation location on which to perform the first  
uninterruptible operation, responsive to whether the system is being used to perform a  
subtraction or addition.